

CLAIMS

What is claimed is:

- Sub
Pat
1. A method of creating run time executable code, comprising:
partitioning a processing element array into a plurality of hardware
accelerators;
decomposing a program description into a plurality of kernel sections;
mapping said kernel sections into a plurality of hardware dependent
designs; and
forming a matrix describing said hardware accelerators and said designs
configured to support run time execution.
2. The method of claim 1, wherein said partitioning includes partitioning
into digital signal processors.
3. The method of claim 1, wherein said partitioning includes partitioning
into bins.
4. The method of claim 1, wherein said mapping includes mapping into
multiple hardware contexts.

1 5. The method of claim 4, wherein said mapping into multiple hardware
2 contexts includes mapping a first set of variants.

1 6. The method of claim 5, wherein said first set of variants are produced
2 based upon resource usage.

1 7. The method of claim 5, wherein said mapping includes mapping a second
2 set of variants of said designs configured to support multiple hardware
3 configurations of one of a plurality of bins.

1 8. The method of claim 1, wherein said mapping is performed by a place
2 and route.

1 9. The method of claim 1, wherein said decomposing is performed
2 manually.

1 10. The method of claim 1, wherein said decomposing is performed by a
2 software profiler.

1 11. The method of claim 10, wherein said decomposing includes executing
2 code compiled from said program description and monitoring timing of said
3 executing.

Handwritten initials "JH" and "GC" with a checkmark.

00000000:5245960

- 1 20. The method of claim 1, wherein said matrix is sparsely-populated.
- 1 21. The method of claim 1, wherein said matrix is fully-populated.
- 1 22. A system for creating run time executable code, comprising:
 - 2 a plurality of hardware accelerators partitioned from a processing
 - 3 element array;
 - 4 a plurality of kernel sections created from a program description;
 - 5 a plurality of hardware dependent designs derived from said kernel
 - 6 sections; and
 - 7 a matrix describing said hardware accelerators and said designs
 - 8 configured to support run time execution.
- 1 23. The system of claim 22, wherein said hardware accelerators includes
- 2 digital signal processors.
- 1 24. The system of claim 22, wherein said hardware accelerators includes
- 2 bins.
- 1 25. The system of claim 24, wherein said bins support multiple hardware
- 2 contexts.

1 33. The system of claim 32, wherein said software profiler includes a set of
2 test data.

1 34. The system of claim 32, wherein said software profiler determines
2 functions that consume a significant portion of said time consumed.

1 35. The system of claim 31, wherein said software profiler is configured to
2 identify kernel sections by identifying regular structures.

1 36. The system of claim 31, wherein said software profiler is configured to
2 identify kernel sections by identifying sections with a limited number of inputs
3 and outputs.

1 37. The system of claim 31, wherein said software profiler is configured to
2 identify kernel sections by identifying sections with a limited number of
3 branches.

1 38. The system of claim 31, wherein said profiler identifies overhead
2 sections.

1 39. The system of claim 22, wherein said designs include microcode.

1 40. The system of claim 39, wherein said microcode includes context
2 dependent configurations.

1 41. The system of claim 22, wherein said matrix is sparsely-populated.

1 42. The system of claim 22, wherein said matrix is fully-populated.

1 43. A machine-readable medium having stored thereon instructions for
2 processing elements, which when executed by said processing elements
3 perform the following:

4 partitioning a processing element array into a plurality of hardware
5 accelerators;

6 decomposing a program description into a plurality of kernel sections;

7 mapping said kernel sections into a plurality of hardware dependent
8 designs; and

9 forming a matrix describing said hardware accelerators and said designs
10 configured to support run time execution.

Sub
Pat

- 1 44. A system configured to create run time executable code, comprising:
2 means for partitioning a processing element array into a plurality of
3 hardware accelerators;
4 means for decomposing a program description into a plurality of kernel
5 sections;
6 means for mapping said kernel sections into a plurality of hardware
7 dependent designs; and
8 means for forming a matrix describing said hardware accelerators and
9 said designs configured to support run time execution.

000000-000000